Characterizing Solution Concepts in Games Using Knowledge-Based Programs

Joseph Y. Halpern*
Computer Science Department
Cornell University, U.S.A.
e-mail: halpern@cs.cornell.edu

Yoram Moses
Department of Electrical Engineering
Technion—Israel Institute of Technology
32000 Haifa, Israel
email: moses@ee.technion.ac.il

Abstract

We show how solution concepts in games such as Nash equilibrium, correlated equilibrium, rationalizability, and sequential equilibrium can be given a uniform definition in terms of *knowledge-based programs*. Intuitively, all solution concepts are implementations of two knowledge-based programs, one appropriate for games represented in normal form, the other for games represented in extensive form. These knowledge-based programs can be viewed as embodying rationality. The representation works even if (a) information sets do not capture an agent's knowledge, (b) uncertainty is not represented by probability, or (c) the underlying game is not common knowledge.

1 Introduction

Game theorists represent games in two standard ways: in *normal form*, where each agent simply chooses a strategy, and in *extensive form*, using game trees, where the agents make choices over time. An extensive-form representation has the advantage that it describes the dynamic structure of the game—it explicitly represents the sequence of decision problems encountered by agents. However, the extensive-form representation purports to do more than just describe the structure of the game; it also attempts to represent the information that players have in the game, by the use of *information sets*. Intuitively, an information set consists of a set of nodes in the game tree where a player has the same information. However, as Halpern [1997] has pointed out, information sets may not adequately represent a player's information.

Halpern makes this point by considering the following single-agent game of imperfect recall, originally presented by Piccione and Rubinstein [1997]: The game starts with nature moving either left or right, each with probability $1/2$. The agent can then either stop the game (playing move S) and get a payoff of 2, or continue, by playing move B. If he continues, he gets a high payoff if he matches nature's move, and a low payoff otherwise. Although he originally knows nature's move, the information set that includes the nodes labeled x_3 and x_4 is intended to indicate that the player forgets whether nature moved left or right after moving B. Intuitively, when he is at the information set X, the agent is not supposed to know whether he is at x_3 or at x_4.

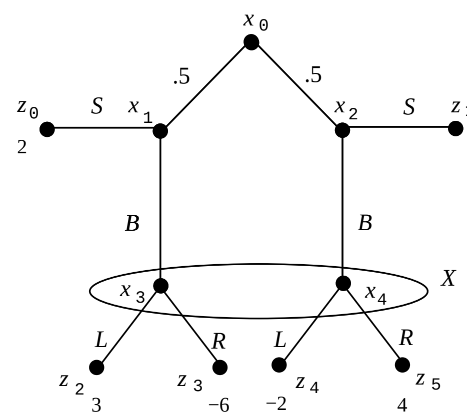

Figure 1: A game of imperfect recall.

It is not hard to show that the strategy that maximizes expected utility chooses action S at node x_1, action B at node x_2, and action R at the information set X consisting of x_3 and x_4. Call this strategy f. Let f' be the strategy of choosing action B at x_1, action S at x_2, and L at X. Piccione and Rubinstein argue that if node x_1 is reached, the player should reconsider, and decide to switch from f to f'. As Halpern points out, this is indeed true, provided that the player knows at each stage of the game what strategy he is currently using. However, in that case, if the player is using f at the information set, then he knows that he is at node x_4; if he has switched and is using f', then he knows that he is at x_3. So, in this setting, it is no longer the case that the player does not know whether he is at x_3 or x_4 in the information set; he can infer which state he is at from the strategy he is using.

In game theory, a *strategy* is taken to be a function from information sets to actions. The intuition behind this is that, since an agent cannot tell the nodes in an information set

*Supported in part by NSF under grants CTC-0208535 and ITR-0325453, by ONR under grant N00014-02-1-0455, by the DoD Multidisciplinary University Research Initiative (MURI) program administered by the ONR under grants N00014-01-1-0795 and N00014-04-1-0725, and by AFOSR under grant F49620-02-1-0101.

apart, he must do the same thing at all these nodes. But this example shows that if the agent has imperfect recall but can switch strategies, then he can arrange to do different things at different nodes in the same information set. As Halpern [1997] observes, ' "situations that [an agent] cannot distinguish" and "nodes in the same information set" may be two quite different notions.' He suggests using the game tree to describe the structure of the game, and using the runs and systems framework [Fagin *et al.*, 1995] to describe the agent's information. The idea is that an agent has an internal *local state* that describes all the information that he has. A strategy (or *protocol* in the language of [Fagin *et al.*, 1995]) is a function from local states to actions. Protocols capture the intuition that what an agent does can depend only what he knows. But now an agent's knowledge is represented by its local state, not by an information set. Different assumptions about what agents know (for example, whether they know their current strategies) are captured by running the same protocol in different *contexts*. If the information sets appropriately represent an agent's knowledge in a game, then we can identify local states with information sets. But, as the example above shows, we cannot do this in general.

A number of *solution concepts* have been considered in the game-theory literature, ranging from Nash equilibrium and *correlated equilibrium* to refinements of Nash equilibrium such as *sequential equilibrium* and weaker notions such as *rationalizability* (see [Osborne and Rubinstein, 1994] for an overview). The fact that game trees represent both the game and the players' information has proved critical in defining solution concepts in extensive-form games. Can we still represent solution concepts in a useful way using runs and systems to represent a player's information? As we show here, not only can we do this, but we can do it in a way that gives deeper insight into solution concepts. Indeed, all the standard solution concepts in the literature can be understood as instances of a single *knowledge-based (kb) program* [Fagin *et al.*, 1995; 1997], which captures the underlying intuition that a player should make a best response, given her beliefs. The differences between solution concepts arise from running the kb program in different contexts.

In a kb program, a player's actions depend explicitly on the player's knowledge. For example, a kb program could have a test that says "If you don't know that Ann received the information, then send her a message", which can be written

$$\textbf{if } \neg B_i(\text{Ann received info}) \textbf{ then } \text{send Ann a message.}$$

This kb program has the form of a standard **if** ... **then** statement, except that the test in the **if** clause is a test on i's knowledge (expressed using the modal operator B_i for belief; see Section 2 for a discussion of the use of knowledge vs. belief).

Using such tests for knowledge allows us to abstract away from low-level details of how the knowledge is obtained. Kb programs have been applied to a number of problems in the computer science literature (see [Fagin *et al.*, 1995] and the references therein). To see how they can be applied to understand equilibrium, given a game Γ in normal form, let $\mathcal{S}_i(\Gamma)$ consist of all the pure strategies for player i in Γ. Roughly speaking, we want a kb program that says that if player i believes that she is about to perform strategy S (which we express with the formula $\mathsf{do}_i(S)$), and she believes that she would not do any better with another strategy, then she should indeed go ahead and run S. This test can be viewed as embodying rationality. There is a subtlety in expressing the statement "she would not do any better with another strategy". We express this by saying "if her expected utility, given that she will use strategy S, is x, then her expected utility if she were to use strategy S' is at most x." The "if she were to use S'" is a *counterfactual* statement. She is planning to use strategy S, but is contemplating what would happen if she were to do something counter to fact, namely, to use S'. Counterfactuals have been the subject of intense study in the philosophy literature (see, for example, [Lewis, 1973; Stalnaker, 1968]) and, more recently, in the game theory literature (see, for example, [Aumann, 1995; Halpern, 2001; Samet, 1996]). We write the counterfactual "If A were the case then B would be true" as "$A \succeq B$". Although this statement involves an "if ... then", the semantics of the counterfactual implication $A \succeq B$ is quite different from the material implication $A \Rightarrow B$. In particular, while $A \Rightarrow B$ is true if A is false, $A \succeq B$ might not be.

With this background, consider the following kb program for player i:

$$\begin{array}{l} \textbf{for each } \text{strategy } S \in \mathcal{S}_i(\Gamma) \textbf{ do} \\ \quad \textbf{if } B_i(\mathsf{do}_i(S) \wedge \forall x(\mathrm{EU}_i = x \Rightarrow \\ \qquad \bigwedge_{S' \in \mathcal{S}_i(\Gamma)}(\mathsf{do}_i(S') \succeq (\mathrm{EU}_i \leq x)))) \textbf{ then } S. \end{array}$$

This kb program is meant to capture the intuition above. Intuitively, it says that if player i believes that she is about to perform strategy S and, if her expected utility is x, then if she were to perform another strategy S', then her expected utility would be no greater than x, then she should perform strategy S. Call this kb program $\mathbf{EQNF}^{\Gamma}$ (with the individual instance for player i denoted by $\mathbf{EQNF}_i^{\Gamma}$). As we show, if all players follow $\mathbf{EQNF}^{\Gamma}$, then they end up playing some type of equilibrium. Which type of equilibrium they play depends on the context. Due to space considerations, we focus on three examples in this abstract. If the players have a common prior on the joint strategies being used, and this common prior is such that players' beliefs are independent of the strategies they use, then they play a Nash equilibrium. Without this independence assumption, we get a *correlated equilibrium*. On the other hand, if players have possibly different priors on the space of strategies, then this kb program defines *rationalizable* strategies [Bernheim, 1984; Pearce, 1984].

To deal with extensive-form games, we need a slightly different kb program, since agents choose moves, not strategies. Let $\mathbf{EQEF}_i^{\Gamma}$ be the following program, where $\mathsf{a} \in PM$ denotes that a is a move that is currently possible.

$$\begin{array}{l} \textbf{for each } \text{move } \mathsf{a} \in PM \textbf{ do} \\ \quad \textbf{if } B_i(\mathsf{do}_i(\mathsf{a}) \wedge \forall x((\mathrm{EU}_i = x) \Rightarrow \\ \qquad \bigwedge_{\mathsf{a}' \in PM}(\mathsf{do}_i(\mathsf{a}') \succeq (\mathrm{EU}_i \leq x)))) \textbf{ then } \mathsf{a}. \end{array}$$

Just as $\mathbf{EQNF}^{\Gamma}$ characterizes equilibria of a game Γ represented in normal form, $\mathbf{EQEF}^{\Gamma}$ characterizes equilibria of a game represented in extensive form. We give one example here: sequential equilibrium. To capture sequential equilibrium, we need to assume that information sets do correctly

describe an agent's knowledge. If we drop this assumption, however, we can distinguish between the two equilibria for the game described in Figure 1.

All these solution concepts are based on expected utility. But we can also consider solution concepts based on other decision rules. For example, Boutilier and Hyafil [2004] consider *minimax-regret* equilibria, where each player uses a strategy that is a best-response in a minimax-regret sense to the choices of the other players. Similarly, we can use *maximin equilibria* [Aghassi and Bertsimas, 2006]. As pointed out by Chu and Halpern [2003], all these decision rules can be viewed as instances of a generalized notion of expected utility, where uncertainty is represented by a *plausibility measure*, a generalization of a probability measure, utilities are elements of an arbitrary partially ordered space, and plausibilities and utilities are combined using $\oplus$ and $\otimes$, generalizations of $+$ and $\times$. We show in the full paper that, just by interpreting "$\mathrm{EU}_i = u$" appropriately, we can capture these more exotic solution concepts as well. Moreover, we can capture solution concepts in games where the game itself is not common knowledge, or where agents are not aware of all moves available, as discussed by Halpern and Rêgo [2006].

Our approach thus provides a powerful tool for representing solution concepts, which works even if (a) information sets do not capture an agent's knowledge, (b) uncertainty is not represented by probability, or (c) the underlying game is not common knowledge.

The rest of this paper is organized as follows. In Section 2, we review the relevant background on game theory and knowledge-based programs. In Section 3, we show that $\mathbf{EQNF}^{\Gamma}$ and $\mathbf{EQEF}^{\Gamma}$ characterize Nash equilibrium, correlated equilibrium, rationalizability, and sequential equilibrium in a game Γ in the appropriate contexts. We conclude in Section 4 with a discussion of how our results compare to other characterizations of solution concepts.

2 Background

In this section, we review the relevant background on games and knowledge-based programs. We describe only what we need for proving our results. The reader is encouraged to consult [Osborne and Rubinstein, 1994] for more on game theory, [Fagin *et al.*, 1995; 1997] for more on knowledge-based programs without counterfactuals, and [Halpern and Moses, 2004] for more on adding counterfactuals to knowledge-based programs.

2.1 Games and Strategies

A game in *extensive form* is described by a game tree. Associated with each non-leaf node or history is either a player—the player whose move it is at that node—or nature (which can make a randomized move). The nodes where a player i moves are further partitioned into *information sets*. With each run or maximal history h in the game tree and player i we can associate i's utility, denoted $u_i(h)$, if that run is played. A *strategy* for player i is a (possibly randomized) function from i's information sets to actions. Thus a strategy for player i tells player i what to do at each node in the game tree where i is supposed to move. Intuitively, at all the nodes that player i cannot tell apart, player i must do the same thing. A *joint strategy* $\vec{S} = (S_1, \ldots, S_n)$ for the players determines a distribution over paths in the game tree. A *normal-form game* can be viewed as a special case of an extensive-form game where each player makes only one move, and all players move simultaneously.

2.2 Protocols, Systems, and Contexts

To explain kb programs, we must first describe standard protocols. We assume that, at any given point in time, a player in a game is in some *local state*. The local state could include the history of the game up to this point, the strategy being used by the player, and perhaps some other features of the player's type, such as beliefs about the strategies being used by other players. A global state is a tuple consisting of a local state for each player.

A *protocol* for player i is a function from player i's local states to actions. For ease of exposition, we consider only deterministic protocols, although it is relatively straightforward to model randomized protocols—corresponding to mixed strategies—as functions from local states to distributions over actions. Although we restrict to deterministic protocols, we deal with mixed strategies by considering distributions over pure strategies.

A *run* is a sequence of global states; formally, a run is a function from times to global states. Thus, $r(m)$ is the global state in run r at time m. A *point* is a pair (r, m) consisting of a run r and time m. Let $r_i(m)$ be i's local state at the point (r, m); that is, if $r(m) = (s_1, \ldots, s_n)$, then $r_i(m) = s_i$. A *joint protocol* is an assignment of a protocol for each player; essentially, a joint protocol is a joint strategy. At each point, a joint protocol $\vec{P}$ performs a *joint action* $(P_1(r_1(m)), \ldots, P_n(r_n(m)))$, which changes the global state. Thus, given an initial global state, a joint protocol $\vec{P}$ generates a (unique) run, which can be thought of as an execution of $\vec{P}$. The runs in a normal-form game involve only one round and two time steps: time 0 (the initial state) and time 1, after the joint strategy has been executed. (We assume that the payoff is then represented in the player's local state at time 1.) In an extensive-form game, a run is again characterized by the strategies used, but now the length of the run depends on the path of play.

A *probabilistic system* is a tuple $\mathcal{PS} = (\mathcal{R}, \vec{\mu})$, where $\mathcal{R}$ is a set of runs and $\vec{\mu} = (\mu_1, \ldots, \mu_n)$ associates a probablity μ_i on the runs of $\mathcal{R}$ with each player i. Intuitively, μ_i represents player i's prior beliefs. In the special case where $\mu_1 = \cdots = \mu_n = \mu$, the players have a *common prior* μ on $\mathcal{R}$. In this case, we write just $(\mathcal{R}, \mu)$.

We are interested in the system corresponding to a joint protocol $\vec{P}$. To determine this system, we need to describe the setting in which $\vec{P}$ is being executed. For our purposes, this setting can be modeled by a set $\mathcal{G}$ of global states, a subset $\mathcal{G}_0$ of $\mathcal{G}$ that describes the possible *initial* global states, a set $\mathcal{A}_s$ of possible joint actions at each global state s, and n probability measures on $\mathcal{G}_0$, one for each player. Thus, a *probabilistic context* is a tuple $\gamma = (\mathcal{G}, \mathcal{G}_0, \{\mathcal{A}_s : s \in \mathcal{G}\}, \vec{\mu})$.[1] A joint

[1] We are implicitly assuming that the global state that results from

protocol $\vec{P}$ is *appropriate* for such a context γ if, for every global state s, the joint actions that $\vec{P}$ can generate are in $\mathcal{A}_s$. When $\vec{P}$ is appropriate for γ, we abuse notation slightly and refer to γ by specifying only the pair $(\mathcal{G}_0, \vec{\mu})$. A protocol $\vec{P}$ and a context γ for which $\vec{P}$ is appropriate generate a system; the system depends on the initial states and probability measures in γ. Since these are all that matter, we typically simplify the description of a context by omitting the set $\mathcal{G}$ of global states and the sets $\mathcal{A}_s$ of global actions. Let $\mathbf{R}(\vec{P},\gamma)$ denote the system generated by joint protocol $\vec{P}$ in context γ. If $\gamma = (\mathcal{G}_0, \vec{\mu})$, then $\mathbf{R}(\vec{P},\gamma) = (\mathcal{R}, \vec{\mu}')$, where $\mathcal{R}$ consists of a the run $r_{\vec{s}}$ for each initial state $\vec{s} \in \mathcal{G}_0$, where $r_{\vec{s}}$ is the run generated by $\vec{P}$ when started in state $\vec{s}$, and $\mu'_i(r_{\vec{s}}) = \mu_i(\vec{s})$, for $i = 1,\ldots,n$.

A probabilistic system $(\mathcal{R}, \vec{\mu}')$ is *compatible* with a context $\gamma = (\mathcal{G}_0, \vec{\mu})$ if (a) every initial state in $\mathcal{G}_0$ is the initial state of some run in $\mathcal{R}$, (b) every run is the run of some protocol appropriate for γ, and (c) if $\mathcal{R}(\vec{s})$ is the set of runs in $\mathcal{R}$ with initial global state $\vec{s}$, then $\mu'_j(\mathcal{R}(\vec{s})) = \mu_j(\vec{s})$, for $j = 1,\ldots,n$. Clearly $\mathbf{R}(\vec{P},\gamma)$ is compatible with γ.

We can think of the context as describing background information. In distributed-systems applications, the context also typically includes information about message delivery. For example, it may determine whether all messages sent are received in one round, or whether they may take up to, say, five rounds. Moreover, when this is not obvious, the context specifies how actions transform the global state; for example, it describes what happens if in the same joint action two players attempt to modify the same memory cell. Since such issues do not arise in the games we consider, we ignore these facets of contexts here. For simplicity, we consider only contexts where each initial state corresponds to a particular joint strategy of Γ. That is, Σ_i^Γ is a set of local states for player i indexed by (pure) strategies. The set Σ_i^Γ can be viewed as describing i's types; the state s_S can the thought of as the initial state where player i's type is such that he plays S (although we stress that this is only intuition; player i does not *have* to play S at the state s_S). Let $\mathcal{G}_0^\Gamma = \Sigma_1^\Gamma \times \ldots \times \Sigma_n^\Gamma$. We will be interested in contexts where the set of initial global states is a subset $\mathcal{G}_0$ of $\mathcal{G}_0^\Gamma$. In a normal-form game, the only actions possible for player i at an initial global state amount to choosing a pure strategy, so the joint actions are joint strategies; no actions are possible at later times. For an extensive-form game, the possible moves are described by the game tree. We say that a context for an extensive-form game is *standard* if the local states have the form (s, I), where s is the initial state and I is the current information set. In a standard context, an agent's knowledge is indeed described by the information set. However, we do not require a context to be standard. For example, if an agent is allowed to switch strategies, then the local state could include the history of strategies used. In such a context, the agent in the game of Figure 1 would know more than just what is in the information set, and would want to switch strategies.

performing a joint action in $\mathcal{A}_s$ at the global state s is unique and obvious; otherwise, such information would also appear in the context, as in the general framework of [Fagin *et al.*, 1995].

2.3 Knowledge-Based Programs

A *knowledge-based program* is a syntactic object. For our purposes, we can take a knowledge-based program for player i to have the form

$$\begin{array}{l}\textbf{if } \kappa_1 \textbf{ then } \mathsf{a}_1\\ \textbf{if } \kappa_2 \textbf{ then } \mathsf{a}_2\\ \ldots,\end{array}$$

where each κ_j is a Boolean combination of formulas of the form $B_i\varphi$, in which the φ's can have nested occurrences of B_ℓ operators and counterfactual implications. We assume that the tests $\kappa_1, \kappa_2, \ldots$ are mutually exclusive and exhaustive, so that exactly one will evaluate to true in any given instance. The program $\mathbf{EQNF}_i^\Gamma$ can be written in this form by simply replacing the $\mathbf{for}\ \ldots\ \mathbf{do}$ statement by one line for each pure strategy in $\mathcal{S}_i(\Gamma)$; similarly for $\mathbf{EQEF}_i^\Gamma$.

We want to associate a protocol with a kb program. Unfortunately, we cannot "execute" a kb program as we can a protocol. How the kb program executes depends on the outcome of tests κ_j. Since the tests involve beliefs and counterfactuals, we need to interpret them with respect to a system. The idea is that a kb program Pg_i for player i and a probabilistic system $\mathcal{PS}$ together determine a protocol P for player i. Rather than giving the general definitions (which can be found in [Halpern and Moses, 2004]), we just show how they work in the two kb programs we consider in this paper: $\mathbf{EQNF}$ and $\mathbf{EQEF}$.

Given a system $\mathcal{PS} = (\mathcal{R}, \vec{\mu})$, we associate with each formula φ a set $[\![\varphi]\!]_{\mathcal{PS}}$ of points in $\mathcal{PS}$. Intuitively, $[\![\varphi]\!]_{\mathcal{PS}}$ is the set of points of $\mathcal{PS}$ where the formula φ is true. We need a little notation:

- If E is a set of points in $\mathcal{PS}$, let $\mathcal{R}(E)$ denote the set of runs going through points in E; that is $\mathcal{R}(E) = \{r : \exists m((r,m) \in E)\}$.
- Let $\mathcal{K}_i(r,m)$ denote the set of points that i cannot distinguish from (r,m): $\mathcal{K}_i(r,m) = \{(r',m') : (r'_i(m') = r_i(m)\}$. Roughly speaking, $\mathcal{K}_i(r,m)$ corresponds to i's information set at the point (r,m).
- Given a point (r,m) and a player i, let $\mu_{(i,r,m)}$ be the probability measure that results from conditioning μ^i on $\mathcal{K}_i(r,m)$, i's information at (r,m). We cannot condition on $\mathcal{K}_i(r,m)$ directly: μ^i is a probability measure on runs, and $\mathcal{K}_i(r,m)$ is a set of points. So we actually condition, not on $\mathcal{K}_i(r,m)$, but on $\mathcal{R}(\mathcal{K}_i(r,m))$, the set of runs going through the points in $\mathcal{K}_i(r,m)$. Thus, $\mu_{i,r,m} = \mu^i \mid \mathcal{R}(\mathcal{K}_i(r,m))$. (For the purposes of this abstract, we do not specify $\mu_{i,r,m}$ if $\mu_i(\mathcal{R}(\mathcal{K}_i(r,m))) = 0$. It turns out not to be relevant to our discussion.)

The kb programs we consider in this paper use a limited collection of formulas. We now can define $[\![\varphi]\!]_{\mathcal{PS}}$ for the formulas we consider that do not involve counterfactuals.

- In a system $\mathcal{PS}$ corresponding to a normal-form game Γ, if $S \in \mathcal{S}_i(\Gamma)$, then $[\![\mathsf{do}_i(S)]\!]_{\mathcal{PS}}$ is the set of initial points $(r,0)$ such that player i uses strategy S in run r.
- Similarly, if $\mathcal{PS}$ corresponds to an extensive-form game, then $[\![\mathsf{do}_i(\mathsf{a})]\!]_{\mathcal{PS}}$ is the set of points (r,m) of $\mathcal{PS}$ at which i performs action a.

- Player i believes a formula φ at a point (r,m) if the event corresponding to formula φ has probability 1 according to $\mu_{i,r,m}$. That is, $(r,m) \in [\![B_i\varphi]\!]_{\mathcal{PS}}$ if $\mu_i(\mathcal{R}(\mathcal{K}_i(r,m)) \neq 0$ (so that conditioning on $\mathcal{K}_i(r,m)$ is defined) and $\mu_{i,r,m}([\![\varphi]\!]_{\mathcal{PS}} \cap \mathcal{K}_i(r,m)) = 1$.
- With every run r in the systems we consider, we can associate the joint (pure) strategy $\vec{S}$ used in r.[2] This pure strategy determines the history in the game, and thus determines player i's utility. Thus, we can associate with every point (r,m) player i's *expected* utility at (r,m), where the expectation is taken with respect to the probability $\mu_{i,r,m}$. If u is a real number, then $[\![\mathrm{EU}_i = u]\!]_{\mathcal{PS}}$ is the set of points where player i's expected utility is u; $[\![\mathrm{EU}_i \leq u]\!]_{\mathcal{PS}}$ is defined similarly.
- Assume that $\varphi(x)$ has no occurrences of $\forall$. Then $[\![\forall x\varphi(x))]\!]_{\mathcal{PS}} = \cap_{a\in I\!R}[\![\varphi[x/a]]\!]_{\mathcal{PS}}$, where $\varphi[x/a]$ is the result of replacing all occurrences of x in φ by a. That is, $\forall x$ is just universal quantification over x, where x ranges over the reals. This quantification arises for us when x represents a utility, so that $\forall x\varphi(x)$ is saying that φ holds for all choices of utility.

We now give the semantics of formulas involving counterfactuals. Here we consider only a restricted class of such formulas, those where the counterfactual only occurs in the form $\mathsf{do}_i(S) \succeq \varphi$, which should be read as "if i were to use strategy S, then φ would be true. Intuitively, $\mathsf{do}_i(S) \succeq \varphi$ is true at a point (r,m) if φ holds in a world that differs from (r,m) only in that i uses the strategy S. That is, $\mathsf{do}_i(S) \succeq \varphi$ is true at (r,m) if φ is true at the point (r',m) where, in run r', player i uses strategy S and all the other players use the same strategy that they do at (r,m). (This can be viewed as an instance of the general semantics for counterfactuals used in the philosophy literature [Lewis, 1973; Stalnaker, 1968] where $\psi \succeq \varphi$ is taken to be true at a world w if φ is true at all the worlds w' closest to w where ψ is true.) Of course, if i actually uses strategy S in run r, then $r' = r$. Similarly, in an extensive-form game Γ, the closest point to (r,m) where $\mathsf{do}_i(\mathsf{a}')$ is true (assuming that a' is an action that i can perform in the local state $r_i(m)$) is the point (r',m) where all players other than player i use the same protocol in r' and r, and i's protocol in r' agrees with i's protocol in r except at the local state $r_i(m)$, where i performs action a'. Thus, r' is the run that results from player i making a single deviation (to a' at time m) from the protocol she uses in r, and all other players use the same protocol as in r.

There is a problem with this approach. There is no guarantee that, in general, such a closest point (r',m) exists in the system $\mathcal{PS}$. To deal with this problem, we restrict attention to a class of systems where this point is guaranteed to exist. A system $(\mathcal{R},\vec{\mu})$ is *complete with respect to context* γ if $\mathcal{R}$ includes every run generated by a protocol appropriate for context γ. In complete systems, the closest point (r',m) is guaranteed to exist. For the remainder of the paper, we evaluate formulas only with respect to complete systems. In a complete system $\mathcal{PS}$, we define $[\![\mathsf{do}_i(S) \succeq \varphi]\!]_{\mathcal{PS}}$ to consist of all the points (r,m) such that the closest point (r',m) to (r,m) where i uses strategy S is in $[\![\varphi]\!]_{\mathcal{PS}}$. The definition of $[\![\mathsf{do}_i(\mathsf{a}) \succeq \varphi]\!]_{\mathcal{PS}}$ is similar. We say that a complete system $(\mathcal{R}',\vec{\mu}')$ *extends* $(\mathcal{R},\vec{\mu})$ if μ_j and μ'_j agree on $\mathcal{R}$ (so that $\mu'_j(A) = \mu_j(A)$) for all $A \subseteq \mathcal{R}$) for $j = 1,\ldots,n$.

Since each formula κ that appears as a test in a kb program Pg_i for player i is a Boolean combination of formulas of the form $B_i\varphi$, it is easy to check that if $(r,m) \in [\![\kappa]\!]_{\mathcal{PS}}$, then $\mathcal{K}_i(r,m) \subseteq [\![\kappa]\!]_{\mathcal{PS}}$. In other words, the truth of κ depends only on i's local state. Moreover, since the tests are mutually exclusive and exhaustive, exactly one of them holds in each local state. Given a system $\mathcal{PS}$, we take the protocol $\mathsf{Pg}_i^{\mathcal{PS}}$ to be such that $\mathsf{Pg}_i^{\mathcal{PS}}(\ell) = \mathsf{a}_j$ if, for some point (r,m) in $\mathcal{PS}$ with $r_i(m) = \ell$, we have $(r,m) \in [\![\kappa_j]\!]_{\mathcal{PS}}$. Since $\kappa_1,\kappa_2,\ldots$ are mutually exclusive and exhaustive, there is exactly one action a_j with this property.

We are mainly interested in protocols that implement a kb program. Intuitively, a joint protocol $\vec{P}$ implements a kb program $\vec{\mathsf{Pg}}$ in context γ if $\vec{P}$ performs the same actions as $\vec{\mathsf{Pg}}$ in all runs of $\vec{P}$ that have positive probability, assuming that the knowledge tests in $\vec{\mathsf{Pg}}$ are interpreted with respect to the complete system $\mathcal{PS}$ extending $\mathbf{R}(\vec{P},\gamma)$. Formally, a joint protocol $\vec{P}$ *(de facto) implements* a joint kb program $\vec{\mathsf{Pg}}$ [Halpern and Moses, 2004] in a context $\gamma = (\mathcal{G}_0,\vec{\mu})$ if $P_i(\ell) = \mathsf{Pg}_i^{\mathcal{PS}}(\ell)$ for every local state $\ell = r_i(m)$ such that $r \in \mathbf{R}(\vec{P},\gamma)$ and $\mu_i(r) \neq 0$, where $\mathcal{PS}$ is the complete system extending $\mathbf{R}(\vec{P},\gamma)$. We remark that, in general, there may not be any joint protocols that implement a kb program in a given context, there may be exactly one, or there may be more than one (see [Fagin *et al.*, 1995] for examples). This is somewhat analogous to the fact that there may not be any equilibrium of a game for some notions of equilibrium, there may be one, or there may be more than one.

3 The Main Results

Fix a game Γ in normal form. Let P_i^{nf} be the protocol that, in initial state $s_S \in \Sigma_i^\Gamma$, chooses strategy S; let $\vec{P}^{nf} = (P_1^{nf},\ldots,P_n^{nf})$. Let STRAT_i be the random variable on initial global states that associates with an initial global state s player i's strategy in r. As we said, Nash equilibrium arises in contexts with a common prior. Suppose that $\gamma = (\mathcal{G}_0,\mu)$ is a context with a common prior. We say that μ is *compatible with* the mixed joint strategy $\vec{S}$ if μ is the probability on pure joint strategies induced by $\vec{S}$ (under the obvious identification of initial global states with joint strategies).

Theorem 3.1: *The joint strategy $\vec{S}$ is a Nash equilibrium of the game Γ iff there is a common prior probability measure μ on $\mathcal{G}_0^\Gamma$ such that $\mathrm{STRAT}_1,\ldots,\mathrm{STRAT}_n$ are independent with respect to μ, μ is compatible with $\vec{S}$, and $\vec{P}^{nf}$ implements $\mathbf{EQNF}^\Gamma$ in the context $(\mathcal{G}_0^\Gamma,\mu)$.*

Proof: Suppose that $\vec{S}$ is a (possibly mixed strategy) Nash equilibrium of the game Γ. Let $\mu_{\vec{S}}$ be the unique probability

[2] If we allow players to change strategies during a run, then we will in general have different joint strategies at each point in a run. For our theorems in the next section, we restrict to contexts where players do not change strategies.

on $\mathcal{G}_0^\Gamma$ compatible with $\vec{S}$. If $\vec{S}$ is played, then the probability of a run where the pure joint strategy $(T_1, \ldots, T_n)$ is played is just the product of the probabilities assigned to T_i by S_i, so $\mathrm{STRAT}_1, \ldots, \mathrm{STRAT}_n$ are independent with respect to $\mu_{\vec{S}}$. To see that $\vec{P}^{nf}$ implements $\mathbf{EQNF}^\Gamma$ in the context $\gamma = (\mathcal{G}_0^\Gamma, \mu_{\vec{S}})$, let $\ell = r_i(0)$ be a local state such that $r = \mathbf{R}(\vec{P}^{nf}, \gamma)$ and $\mu(r) \neq 0$. If $\ell = s_T$, then $P_i^{nf}(\ell) = T$, so T must be in the support of S_i. Thus, T must be a best response to $\vec{S}_{-i}$, the joint strategy where each player $j \neq i$ plays its component of $\vec{S}$. Since i uses strategy T in r, the formula $B_i(\mathsf{do}_i(T'))$ holds at $(r, 0)$ iff $T' = T$. Moreover, since T is a best response, if u is i's expected utility with the joint strategy $\vec{S}$, then for all T', the formula $\mathsf{do}_i(T') \succeq (\mathrm{EU}_i \leq u)$ holds at $(r, 0)$. Thus, $(\mathbf{EQNF}_i^\Gamma)^{\mathcal{PS}}(\ell) = T$, where $\mathcal{PS}$ is the complete system extending $\mathbf{R}(\vec{P}^{nf}, \gamma)$. It follows that $\vec{P}^{nf}$ implements $\mathbf{EQNF}^\Gamma$.

For the converse, suppose that μ is a common prior probability measure on $\mathcal{G}_0^\Gamma$, $\mathrm{STRAT}_1, \ldots, \mathrm{STRAT}_n$ are independent with respect to μ, μ is compatible with $\vec{S}$, and $\vec{P}^{nf}$ implements $\mathbf{EQNF}^\Gamma$ in the context $\gamma = (\mathcal{G}_0^\Gamma, \mu)$. We want to show that $\vec{S}$ is a Nash equilibrium. It suffices to show that each pure strategy T in the support of S_i is a best response to $\vec{S}_{-i}$. Since μ is compatible with $\vec{S}$, there must be a run r such that $\mu(r) > 0$ and $r_i(0) = s_T$ (i.e., player i chooses T in run r). It since $\vec{P}^{nf}$ implements $\mathbf{EQNF}^\Gamma$, and in the context γ, $\mathbf{EQNF}^\Gamma$ ensures that no deviation from T can improve i's expected utility with respect to $\vec{S}_{-i}$, it follows that T is indeed a best response. ∎

As is well known, players can sometimes achieve better outcomes than a Nash equilibrium if they have access to a helpful mediator. Consider the simple 2-player game described in Figure 2, where Alice, the row player, must choose between top and bottom (T and B), while Bob, the column player, must choose between left and right (L and R):

	L	R
T	$(3,3)$	$(1,4)$
B	$(4,1)$	$(0,0)$

Figure 2: A simple 2-player game.

It is not hard to check that the best Nash equilibrium for this game has Alice randomizing between T and B, and Bob randomizing between L and R; this gives each of them expected utility 2. They can do better with a trusted mediator, who makes a recommendation by choosing at random between (T, L), (T, R), and (B, L). This gives each of them expected utility $8/3$. This is a *correlated equilibrium* since, for example, if the mediator chooses (T, L), and thus sends recommendation T to Alice and L to Bob, then Alice considers it equally likely that Bob was told L and R, and thus has no incentive to deviate; similarly, Bob has no incentive to deviate. In general, a distribution μ over pure joint strategies is a correlated equilibrium if players cannot do better than following a mediator's recommendation if a mediator makes recommendations according to μ. (Note that, as in our example, if a mediator chooses a joint strategy $(S_1, \ldots, S_n)$ according to μ, the mediator recommends S_i to player i; player i is not told the joint strategy.) We omit the formal definition of correlated equilibrium (due to Aumman [1974]) here; however, we stress that a correlated equilibrium is a distribution over (pure) joint strategies. We can easily capture correlated equilibrium using $\mathbf{EQNF}$.

Theorem 3.2: *The distribution μ on joint strategies is a correlated equilibrium of the game Γ iff $\vec{P}^{nf}$ implements $\mathbf{EQNF}^\Gamma$ in the context $(\mathcal{G}_0^\Gamma, \mu)$.*

Both Nash equilibrium and correlated equilibrium require a common prior on runs. By dropping this assumption, we get another standard solution concept: *rationalizability* [Bernheim, 1984; Pearce, 1984]. Intuitively, a strategy for player i is rationalizable if it is a best response to some beliefs that player i may have about the strategies that other players are following, assuming that these strategies are themselves best responses to beliefs that the other players have about strategies that other players are following, and so on. To make this precise, we need a little notation. Let $\mathcal{S}_{-i} = \Pi_{j \neq i} \mathcal{S}_j$. Let $u_i(\vec{S})$ denote player i's utility if the strategy tuple $\vec{S}$ is played. We describe player i's beliefs about what strategies the other players are using by a probability μ_i on $\mathcal{S}_{-i}$. A strategy S for player i is a *best response to beliefs described by a probability μ_i on $\mathcal{S}_{-i}(\Gamma)$* if $\sum_{\vec{T} \in \mathcal{S}_{-i}} u_i(S, \vec{T}) \mu_i(\vec{T}) \geq \sum_{\vec{T} \in \mathcal{S}_{-i}} u_i(S', \vec{T}) \mu_i(\vec{T})$ for all $S' \in \mathcal{S}_i$. Following Osborne and Rubinstein [1994], we say that a strategy S for player i in game Γ is *rationalizable* if, for each player j, there is a set $Z_j \subseteq \mathcal{S}_j(\Gamma)$ and, for each strategy $T \in Z_j$, a probability measure $\mu_{j,T}$ on $\mathcal{S}_{-j}(\Gamma)$ whose support is Z_{-j} such that

- $S \in Z_i$; and
- for each player j and strategy $T \in Z_j$, T is a best response to the beliefs $\mu_{j,T}$.

For ease of exposition, we consider only pure rationalizable strategies. This is essentially without loss of generality. It is easy to see that a mixed strategy S for player i is a best response to some beliefs μ_i of player i iff each pure strategy in the support of S is a best response to μ_i. Moreover, we can assume without loss of generality that the support of μ_i consists of only pure joint strategies.

Theorem 3.3: *A pure strategy S for player i is rationalizable iff there exist probability measures $\mu_1, \ldots, \mu_n$, a set $\mathcal{G}_0 \subseteq \mathcal{G}_0^\Gamma$, and a state $\vec{s} \in \mathcal{G}_0$ such that $P_i^{nf}(s_i) = S$ and $\vec{P}^{nf}$ implements $\mathbf{EQNF}^\Gamma$ in the context $(\mathcal{G}_0, \vec{\mu})$.*

Proof: First, suppose that $\vec{P}^{nf}$ implements $\mathbf{EQNF}^\Gamma$ in context $(\mathcal{G}_0, \vec{\mu})$. We show that for each state $\vec{s} \in \mathcal{G}_0$ and player i, the strategy $S_{\vec{s},i} = \vec{P}_i^{nf}(s_i)$ is rationalizable. Let $Z_i = \{S_{\vec{s},i} : \vec{s} \in \mathcal{G}_0\}$. For $S \in Z_i$, let $E(S) = \{\vec{s} \in \mathcal{G}_0 : s_i = s_S\}$; that is, $E(S)$ consists consists of all initial global states where player i's local state is s_S; let $\mu_{i,S} = \mu_i(\cdot \mid E(S))$ (under the obvious identification of global states in $\mathcal{G}_0$ with joint strategies). Since $\vec{P}^{nf}$ implements $\mathbf{EQNF}^\Gamma$, it easily follows that S best response to $\mu_{i,S}$. Hence, all the strategies in Z_i are rationalizable, as desired.

For the converse, let Z_i consist of all the pure rationalizable strategies for player i. It follows from the definition of rationalizability that, for each strategy $S \in Z_i$, there exists a probability measure $\mu_{i,S}$ on Z_{-i} such that S is a best response to $\mu_{i,S}$. For a set Z of strategies, we denote by $\tilde{Z}$ the set $\{s_T : T \in Z\}$. Set $\mathcal{G}_0 = \tilde{Z}_1 \times \ldots \times \tilde{Z}_n$, and choose some measure μ_i on $\mathcal{G}_0$ such that $\mu_i(\cdot \mid E(S)) = \mu_{i,S}$ for all $S \in Z_i$. (We can take $\mu_i = \sum_{S \in Z_i} \alpha_S \mu_{i,S}$, where $\alpha_S \in (0,1)$ and $\sum_{S \in Z_i} \alpha_S = 1$.) Recall that $P_i^{nf}(s_S) = S$ for all states s_S. It immediately follows that, for every rationalizable joint strategy $\vec{S} = (S_1, \ldots, S_n)$, both $\vec{s} = (s_{S_1}, \ldots, s_{S_n}) \in \mathcal{G}_0$, and $\vec{S} = \vec{P}^{nf}(\vec{s})$. Since the states in $\mathcal{G}_0$ all correspond to rationalizable strategies, and by definition of rationalizability each (individual) strategy S_i is a best response to $\mu_{i,S}$, it is easy to check that $\vec{P}^{nf}$ implements $\mathbf{EQNF}^{\Gamma}$ in the context $(\mathcal{G}_0^{\Gamma}, \vec{\mu})$, as desired. ∎

We remark that Osborne and Rubinstein's definition of rationalizability allows $\mu_{j,T}$ to be such that j believes that other players' strategy choices are correlated. In most of the literature, players are assumed to believe that other players' choices are made independently. If we add that requirement, then we must impose the same requirement on the probability measures $\mu_1, \ldots, \mu_n$ in Theorem 3.3.

Up to now we have considered solution concepts for games in normal form. Perhaps the best-known solution concept for games in extensive form is *sequential equilibrium* [Kreps and Wilson, 1982]. Roughly speaking, a joint strategy $\vec{S}$ is a sequential equilibrium if S_i is a best response to $\vec{S}_{-i}$ at *all* information sets, not just the information sets that are reached with positive probability when playing $\vec{S}$. To understand how sequential equilibrium differs from Nash equilibrium, consider the game shown in Figure 3.

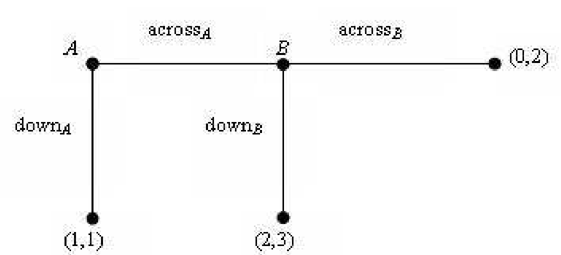

Figure 3: A game with an unreasonable Nash equilibrium.

One Nash equilibrium of this game has A playing down$_A$ and B playing across$_B$. However, this is not a sequential equilibrium, since playing across is not a best response for B if B is called on to play. This is not a problem in a Nash equilibrium because the node where B plays is not reached in the equilibrium. Sequential equilibrium refines Nash equilibrium (in the sense that every sequential equilibrium is a Nash equilibrium) and does not allow solutions such as (down$_A$, across$_B$). Intuitively, in a sequential equilibrium, every player must make a best response at every information set (even if it is reached with probability 0). In the game shown in Figure 3, the unique joint strategy in a sequential equilibrium has A choosing across$_A$ and B choosing down$_B$.

The main difficulty in defining sequential equilibrium lies in capturing the intuition of best response in information sets that are reached with probability 0. To deal with this, a sequential equilibrium is defined to be a pair $(\vec{S}, \beta)$, consisting of a joint strategy $\vec{S}$ and a *belief system* β, which associates with every information set I a probability $\beta(I)$ on the histories in I. There are a number of somewhat subtle consistency conditions on these pairs pairs; we omit them here due to lack of space (see [Kreps and Wilson, 1982; Osborne and Rubinstein, 1994] for details). Our result depends on a recent characterization of sequential equilibrium [Halpern, 2006] that uses nonstandard probabilities, which can assign infinitesimal probabilities to histories. By assuming that every history gets positive (although possibly infinitesimal) probability, we can avoid the problem of dealing with information sets that are reached with probaility 0.

To every nonstandard real number r, there is a closest standard real number denoted *st* (r), and read "the standard part of r": $|r - st\,(r)\,|$ is an infinitesimal. Given a nonstandard probability measure ν, we can define the standard probability measure *st* (ν) by taking $st\,(\nu)\,(w) = st\,(\nu(w))$. A nonstandard probability ν on $\mathcal{G}_0$ is *compatible with* joint strategy $\vec{S}$ if *st* (ν) is the probability on pure strategies induced by $\vec{S}$. When dealing with nonstandard probabilities, we generalize the definition of implementation by requiring only that $\vec{P}$ performs the same actions as $\vec{\mathrm{P}}$g in runs r of $\vec{P}$ such that $st\,(\nu)\,(r) > 0$. Moreover, the expression "$EU_i = x$" in $\mathbf{EQEF}^{\Gamma}$ is interpreted as "the standard part of i's expected utility is x" (since x ranges over the standard real numbers).

Theorem 3.4: *If* Γ *is a game with perfect recall*[3] *there is a belief system* β *such that* $(\vec{S}, \beta)$ *is a sequential equilibrium of* Γ *iff there is a common prior nonstandard probability measure* ν *on* $\mathcal{G}_0^{\Gamma}$ *that gives positive measure to all states such that* $\mathrm{STRAT}_1, \ldots, \mathrm{STRAT}_n$ *are independent with respect to* ν, ν *is compatible with* $\vec{S}$, *and* $\vec{P}^{ef}$ *implements* $\mathbf{EQEF}^{\Gamma}$ *in the standard context* $(\mathcal{G}_0^{\Gamma}, \nu)$.

This is very similar in spirit to Theorem 3.1. The key difference is the use of a nonstandard probability measure. Intuitively, this forces $\vec{S}$ to be a best response even at information sets that are reached with (standard) probability 0.

The effect of interpreting "$\mathrm{EU}_i = x$" as "the standard part of i's expected utility is x" is that we ignore infinitesimal differences. Thus, for example, the strategy $\vec{P}_i^{ef}(\vec{s_0})$ might not be a best response to $\vec{S}_{-i}$; it might just be an ϵ-best response for some infinitesimal ϵ. As we show in the full paper, it follows from Halpern's [2006] results that we can also obtain a characterization of *(trembling hand) perfect equilibrium* [Selten, 1975], another standard refinement of Nash equilibrium, if we interpret "$\mathrm{EU}_i = x$" as "the expected utility for agent i is x" and allow x to range over the nonstandard reals instead of just the standard reals.

[3]These are games where players remember all actions made and the states they have gone through; we give a formal definition in the full paper. See also [Osborne and Rubinstein, 1994].

4 Conclusions

We have shown how a number of different solution concepts from game theory can be captured by essentially one knowledge-based program, which comes in two variants: one appropriate for normal-form games and one for extensive-form games. The differences between these solution concepts is captured by changes in the context in which the games are played: whether players have a common prior (for Nash equilibrium, correlated equilibrium, and sequential equilibrium) or not (for rationalizability), whether strategies are chosen independently (for Nash equilibrium, sequential equilibrium, and rationalizability) or not (for correlated equilibrium); and whether uncertainty is represented using a standard or non-standard probability measure.

Our results can be viewed as showing that each of these solution concepts *sc* can be characterized in terms of common knowledge of rationality (since the kb programs $\mathbf{EQNF}^{\Gamma}$ and $\mathbf{EQEF}^{\Gamma}$ embody rationality, and we are interested in systems "generated" by these program, so that rationality holds at all states), and common knowledge of some other features X_{sc} captured by the context appropriate for *sc* (e.g., that strategies are chosen independently or that the prior). Roughly speaking, our results say that if X_{sc} is common knowledge in a system, then common knowledge of rationality implies that the strategies used must satisfy solution concept *sc*; conversely, if a joint strategy $\vec{S}$ satisfies *sc*, then there is a system where X_{sc} is common knowledge, rationality is common knowledge, and $\vec{S}$ is being played at some state. Results similar in spirit have been proved for rationalizability [Brandenburger and Dekel, 187] and correlated equilibrium [Aumann, 1987]. Our approach allows us to unify and extend these results and, as suggested in the introduction, applies even to settings where the game is not common knowledge and in settings where uncertainty is not represented by probability. We believe that the approach captures the essence of the intuition that a solution concept should embody common knowledge of rationality.

References

[Aghassi and Bertsimas, 2006] M. Aghassi and D. Bertsimas. Robust game theory. *Mathematical Programming, Series B*, 107(1–2):231–273, 2006.

[Aumann, 1974] R. J. Aumann. Subjectivity and correlation in randomized strategies. *Journal of Mathematical Economics*, 1:67–96, 1974.

[Aumann, 1987] R. J. Aumann. Correlated equilibrium as an expression of Bayesian rationality. *Econometrica*, 55:1–18, 1987.

[Aumann, 1995] R. J. Aumann. Backwards induction and common knowledge of rationality. *Games and Economic Behavior*, 8:6–19, 1995.

[Bernheim, 1984] B. D. Bernheim. Rationalizable strategic behavior. *Econometrica*, 52(4):1007–1028, 1984.

[Boutilier and Hyafil, 2004] C. Boutilier and N. Hyafil. Regret minimizing equilibria and mechanisms for games with strict type uncertainty. In *Proc. Twentieth Conference on Uncertainty in Artificial Intelligence (UAI 2004)*, pages 268–277, 2004.

[Brandenburger and Dekel, 187] A. Brandenburger and E. Dekel. Rationalizability and correlated equilibria. *Econometrica*, 55:1391–1402, 187.

[Chu and Halpern, 2003] F. Chu and J. Y. Halpern. Great expectations. Part I: On the customizability of generalized expected utility. In *Proc. Eighteenth International Joint Conference on Artificial Intelligence (IJCAI '03)*, pages 291–296, 2003.

[Fagin *et al.*, 1995] R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning about Knowledge*. MIT Press, Cambridge, Mass., 1995. A revised paperback edition was published in 2003.

[Fagin *et al.*, 1997] R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. Knowledge-based programs. *Distributed Computing*, 10(4):199–225, 1997.

[Halpern and Moses, 2004] J. Y. Halpern and Y. Moses. Using counterfactuals in knowledge-based programming. *Distributed Computing*, 17(2):91–106, 2004.

[Halpern and Rêgo, 2006] J. Y. Halpern and L. C. Rêgo. Extensive games with possibly unaware players. In *Proc. Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 744–751, 2006.

[Halpern, 1997] J. Y. Halpern. On ambiguities in the interpretation of game trees. *Games and Economic Behavior*, 20:66–96, 1997.

[Halpern, 2001] J. Y. Halpern. Substantive rationality and backward induction. *Games and Economic Behavior*, 37:425–435, 2001.

[Halpern, 2006] J. Y. Halpern. A nonstandard characterization of sequential equilibriu, perfect equilibrium, and proper equilibrium. Unpublished manuscript, 2006.

[Kreps and Wilson, 1982] D. M. Kreps and R. B. Wilson. Sequential equilibria. *Econometrica*, 50:863–894, 1982.

[Lewis, 1973] D. K. Lewis. *Counterfactuals*. Harvard University Press, Cambridge, Mass., 1973.

[Osborne and Rubinstein, 1994] M. J. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, Cambridge, Mass., 1994.

[Pearce, 1984] D. G. Pearce. Rationalizable strategic behavior and the problem of perfection. *Econometrica*, 52(4):1029–1050, 1984.

[Piccione and Rubinstein, 1997] M. Piccione and A. Rubinstein. On the interpretation of decision problems with imperfect recall. *Games and Economic Behavior*, 20(1):3–24, 1997.

[Samet, 1996] D. Samet. Hypothetical knowledge and games with perfect information. *Games and Economic Behavior*, 17:230–251, 1996.

[Selten, 1975] R. Selten. Reexamination of the perfectness concept for equilibrium points in extensive games. *International Journal of Game Theory*, 4:25–55, 1975.

[Stalnaker, 1968] R. C. Stalnaker. A semantic analysis of conditional logic. In N. Rescher, editor, *Studies in Logical Theory*, pages 98–112. Oxford University Press, 1968.